

Práctica empresarial: implementación de desarrollo frontend en sitio web y  
procesamiento de datos para migración de sistemas de información

Karem Giovanna Parra Moreno

Universidad Tecnológica de Pereira  
Facultad de Ingenierías  
Ingeniería de Sistemas y Computación  
Pereira  
2021

Práctica empresarial: implementación de desarrollo frontend en sitio web y  
procesamiento de datos para migración de sistemas de información

Karem Giovanna Parra Moreno

Trabajo de grado presentado como requisito para optar al título de  
Ingeniera de Sistemas y Computación

Luz Estela Valencia Ayala

Docente guía

Universidad Tecnológica de Pereira  
Facultad de Ingenierías  
Ingeniería de Sistemas y Computación  
Pereira  
2021

## **CONTENIDO**

<b>INTRODUCCIÓN</b>	<b>4</b>
<b>PLANTEAMIENTO DEL PROBLEMA</b>	<b>5</b>
<b>OBJETIVO GENERAL</b>	<b>6</b>
<b>OBJETIVOS ESPECÍFICOS</b>	<b>6</b>
<b>MARCO INSTITUCIONAL</b>	<b>7</b>
<b>MARCO TEÓRICO</b>	<b>8</b>
<b>METODOLOGÍA</b>	<b>10</b>
<b>DESARROLLO</b>	<b>13</b>
<b>CONCLUSIONES</b>	<b>31</b>
<b>BIBLIOGRAFÍA</b>	<b>32</b>

## **INTRODUCCIÓN**

La práctica empresarial es desarrollada con el fin de que el estudiante logre aplicar y afianzar los conocimientos teóricos y técnicos adquiridos durante su paso por el programa académico. Las actividades planteadas por el escenario de práctica, además de ayudar con el objetivo académico que persigue el estudiante, contribuyen también con el avance de los proyectos que se tienen en curso dentro de la empresa. En este caso, el estudiante participó activamente durante el desarrollo de la práctica en dos proyectos diferentes. En el primer proyecto (mencionado a partir de ahora como proyecto A) el estudiante participó cinco meses aproximadamente y ejecutó actividades relacionadas con el desarrollo frontend para el sitio web del cliente, el cual cuenta con una línea de negocio relacionada al transporte de pasajeros, carga y mensajería. Para un segundo proyecto (mencionado a partir de ahora como proyecto B) donde el cliente cuenta con una línea de negocio dedicada a la comercialización de energía, el estudiante desarrolló actividades de limpieza y procesamiento de datos durante un mes aproximadamente, siendo este un aporte significativo para la totalidad del proyecto de migración de datos de la empresa.

## **1. PLANTEAMIENTO DEL PROBLEMA**

Para el proyecto A, el escenario de práctica se encarga de realizar desarrollos y mejoras de la experiencia de usuario dentro del sitio web del cliente. Estos desarrollos deben ser desplegados sobre la herramienta Adobe Experience Manager, la cual facilita la gestión del contenido del sitio web para el cliente. Dado que realizar y desplegar estos desarrollos requiere una cantidad de tiempo considerable y teniendo en cuenta que los requerimientos del cliente son solicitados en el menor tiempo posible debido a que su sitio tiene una cantidad significativa de tráfico de usuarios por día, se decide crear un nuevo equipo de trabajo que tendrá como objetivo realizar estos desarrollos y desplegarlos en producción de una manera ágil y efectiva.

Por otro lado, el objetivo del proyecto B es conseguir la migración de la totalidad de los datos que se alojan dentro del sistema de información actual del cliente hacia el nuevo sistema de información de SAP, que es un software de planificación de recursos empresariales (ERP por sus siglas en inglés). Para lograrlo se crea un equipo intermediario que realice la limpieza y procesamiento de los datos entregados por el sistema de información actual, buscando así que los registros que sean entregados se pueda presentar de tal manera que no representen complicaciones a la hora de ser cargados en SAP.

## **2. OBJETIVO GENERAL**

Realizar actividades de desarrollo frontend y procesamiento y limpieza de datos para contribuir al alcance de los objetivos y requerimientos establecidos dentro de los diferentes proyectos con los que cuenta la empresa Accenture en el área de Digital and Technology.

## **3. OBJETIVOS ESPECÍFICOS**

- Mejorar la experiencia de usuario de los clientes por medio de la actualización de componentes del sitio web del cliente.
- Apoyar en la automatización de procesos de diferentes áreas del cliente mediante la herramienta Adobe Campaign.
- Procesar los datos provenientes del sistema de información actual del cliente por medio de actividades de carga, validación, transformación y extracción, para conseguir una estandarización de ellos y migrarlos de manera efectiva al nuevo sistema de información.

#### 4. MARCO INSTITUCIONAL

La empresa donde se desarrolla la práctica, Accenture Limited, es una empresa global de consultoría, dedicada a la prestación de servicios de asesoría especializada para diferentes industrias. Esta compañía fue fundada en 1.989, cuenta con más de 469.000 empleados en la actualidad y tiene presencia a nivel mundial.

Accenture maneja cuatro líneas de trabajo nombradas “Accenture Workforces”, con las cuales realizan la atención a los clientes en áreas de consultoría, tecnología y subcontratación, pero también las realizan para la propia empresa. Las cuatro líneas de trabajo son:

- **Consulting:** enfocada en consultoría de gestión, diseño de proceso de producción y aplicación de tecnologías a los negocios.
- **Operation:** enfocada en subcontratación en áreas de operaciones de negocios, soporte técnico, desarrollo y mantenimiento de aplicaciones.
- **Strategy:** enfocada en el diseño de planes de acción y en el apoyo a actividades de negocio de Accenture, incluyendo seguridad jurídica, de servicios, mercadotecnia y gestión financiera.
- **Digital and Technology:** esta línea de trabajo es la que contiene la mayoría de empleados de la compañía y se enfoca en las habilidades tecnológicas específicas que son necesarias para ejecutar proyectos o acuerdos de subcontratación.

## 5. MARCO TEÓRICO

La práctica fue desarrollada dentro de la línea de negocio Digital and Technology de la empresa y se fueron usadas las siguientes tecnologías y herramientas:

- **JavaScript:** Lenguaje de programación interpretado just-in-time con funciones de primera clase. Es un lenguaje de scripting, es decir, que no necesita ser compilado para su ejecución, lo que permite ciertas ventajas a la hora de la definición y desarrollo del código fuente, ya que en este no es necesario la definición de tipos de variables ni funciones main, la ejecución es realizada por un interprete que lee el archivo de código fuente al momento. Para la ejecución de las tareas de la práctica se desarrollan scripts del lado del cliente, que permiten la manipulación del árbol DOM (Document Object Model por sus siglas en inglés).
- **Jquery:** Es una librería de JavaScript, creada para simplificar la manera de interactuar con los documentos HTML (Lenguaje de Marcas de Hipertexto, del inglés HyperText Markup Language), la manipulación del árbol DOM, el control de eventos y la creación de animaciones e interacción a páginas web. Es un software libre y de código abierto, que ofrece funcionalidades basadas en JavaScript que simplifican el código desarrollado únicamente con código puro de JavaScript.
- **Adobe Experience Manager:** AEM por sus siglas en inglés. Es una plataforma de administración y manejo de contenido web desarrollada por Adobe, con el que se proporciona al cliente la administración de contenido para cualquier tipo de comercio electrónico, como lo es la creación de sitios web y aplicaciones, el manejo de contenido de campañas de marketing y la gestión de activos digitales para publicar en diferentes canales, entre otras funcionalidades.



- **Adobe Target:** Es una solución de Adobe que proporciona todo lo necesario para adaptar, personalizar y delimitar las experiencias de los usuarios dentro de las páginas web.
- **Adobe Campaign Classic:** Es una plataforma de interacción multicanal que ofrece opciones específicas para la gestión y ejecución de campañas. Con esta herramienta es posible crear flujos de trabajo que permitan crear, coordinar y ofrecer campañas dinámicas mediante correo electrónico o canales móviles de manera automatizada.
- **Azure DevOps:** Es un conjunto de herramientas de desarrollo de software que permite que los equipos multidisciplinarios trabajen de un modo eficaz en proyectos de todos los tamaños, brindando la oportunidad de utilizar metodologías ágiles para el desarrollo, planificación y seguimiento de las actividades definidas, utilizando repositorios GIT para trabajo colaborativo y herramientas para la gestión y realización de pruebas de código.
- **PostgreSQL:** Es un sistema gestor de bases de datos, relacional y orientado a objetos. Una de sus principales ventajas es que tiene alta concurrencia, es decir que permite que mientras un proceso escriba dentro de una tabla, otros puedan acceder a ella sin presentarse bloqueado. También cuenta con disparadores y funciones. Los disparadores realizan una actividad en específico con base en un evento que ocurra dentro de la base de datos, y las funciones son bloques de código que se ejecutan en el servidor y pueden servir para crear operaciones básicas de programación como lo son las bifurcaciones o los bucles.

## 6. METODOLOGÍA

Para la obtención de los objetivos se maneja la siguiente metodología:

- **Mejorar la experiencia de usuario de los clientes por medio de la actualización de componentes del sitio web del cliente.**

En primer lugar los requerimientos del cliente serán registrados y atendidos por medio de la aplicación Azure DevOps. Dentro de esta será definido el alcance, el impacto y la prioridad de cada uno de ellos. Seguidamente el equipo de trabajo designará una persona encargada de desarrollar el requerimiento, estimar el tiempo requerido e informar del estado de completitud del mismo. Cuando el desarrollo esté listo, se insertará dentro de la página donde se solicitó por medio de la herramienta Adobe Target en un ambiente pre-productivo o de testing y se asigna la tarea al cliente para que se realicen las pruebas necesarias para verificar que se cumplan todas las funcionalidades requeridas en el desarrollo. En caso de ser fallidas las pruebas, se reasigna la tarea al recurso dentro del equipo para su verificación, pero si el desarrollo es aceptado por el cliente, el ambiente donde se visualiza el desarrollo será cambiado a producción por el recurso encargado, donde deberá ser probado nuevamente por el cliente y si cumple con las especificaciones, se dará por terminada la tarea.

- **Apoyar en la automatización de procesos de diferentes áreas del cliente mediante la herramienta Adobe Campaign.**

Los requerimientos del cliente relacionados a las soluciones proporcionadas por la herramienta Adobe Campaign serán recibidos y atendidos con la aplicación Azure DevOps. Desde allí el cliente definirá el alcance, el impacto y la prioridad de los requerimientos. Una vez realizada la solicitud, el equipo se distribuirá las peticiones y el encargado deberá informar sobre el estado de la tarea y el tiempo estimado que tardará en ser terminada. Dependiendo de si el requerimiento es de

creación o actualización se trabajará de diferentes maneras. Cuando se solicita la actualización o mejora de alguno de los flujos de trabajo ya implementados, se realizará el cambio en ambiente productivo y se procederá a las pruebas de funcionamiento por parte del cliente; si el requerimiento se trata de la creación de un nuevo flujo de trabajo, éste será desarrollado y probado dentro del ambiente pre-productivo y una vez se verifique su correcto funcionamiento, será desarrollado y probado por el cliente en ambiente productivo. Cuando se compruebe que el requerimiento desarrollado funciona en ambiente productivo se dará por completado y terminado.

- **Cargar, validar, transformar y extraer los datos provenientes del sistema de información anterior del cliente, para conseguir una estandarización de los datos y poder ser cargados dentro del nuevo sistema de información SAP.**

Los objetos que deberán ser procesados serán entregados al equipo de trabajo por medio de un servidor compartido. Seguidamente se procederá a realizar los 4 grandes procesos con los registros recibidos: carga, validación, transformación y exportación. Si dentro de los procesos de carga y validación se encuentran una cantidad considerable de errores y no pueden ser cargados algunos registros, se realizará un informe al cliente para que puedan verificar si se pueden corregir errores, enviar el objeto de nuevo o si en definitiva no se puede solucionar esos problemas dentro del ciclo actual y se deberá continuar con el proceso de migración. En caso de poder migrar la totalidad de los datos o de no poder solucionar los errores de carga y validación mencionados anteriormente, se procederá a informar al negocio los resultados de los procesos ejecutados para los objetos enviados. Al finalizar los cuatro macro procesos por los que pasan los objetos, se obtendrá un archivo .csv que será extraído por SAP para su carga dentro de ese sistema de información.



## 7. DESARROLLO

A lo largo de la práctica fueron muchos los desarrollos que se realizaron para la página web del proyecto A y aunque todos los requerimientos solicitados (también llamadas historias de usuario o HU's) son diferentes, todos poseen el mismo flujo de trabajo y las mismas etapas de proceso. A continuación se explican e ilustran cada uno de los pasos realizados a la hora de atender los requerimientos del cliente:

- **Estimación de tiempos y creación de tareas relacionadas con la historia de usuario.**

Antes de que los recursos del equipo comiencen con el desarrollo de los requerimientos solicitados, el cliente crea las HU's por medio de la herramienta Azure DevOps, las cuales son entregadas o asignadas al líder del equipo en primera instancia. Dentro de estas se puede añadir la descripción del cambio que se debe realizar, definiendo las reglas relacionadas a las páginas donde debe aplicar el desarrollo, cómo deberían ser los parámetros actuales del componente que se va a modificar (incluido textos en varios idiomas, tamaño y color del componente, eventos nuevos que debe realizar, etc), datos de prueba para la verificación del funcionamiento del desarrollo y otras notas adicionales que pueden surgir dependiendo del tipo de cambio que se esté solicitando; además, dentro de la HU el cliente también ingresa el tipo de requerimiento y la prioridad que se le debe dar a esa HU dentro del backlog de cada equipo.

Azure DevOps, además de documentar los avances dentro del sprint y permitir una gestión más ágil de los recursos y los requerimientos, permite a todos los integrantes del proyecto verificar en qué punto se encuentran cada una de las HU's que están en curso actualmente. Estos breakpoints se definen dentro de la herramienta como “estados” y, para los desarrollos que se manejaban dentro de la célula Adobe (equipo de trabajo), se utilizaban los siguientes estados (ver figura 7.1):

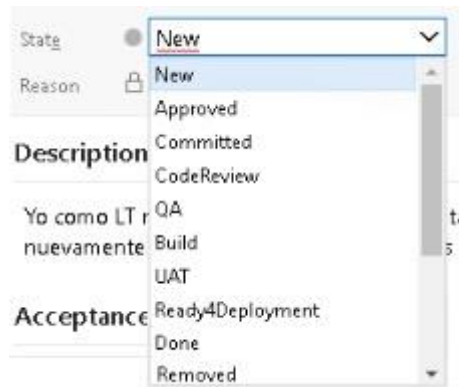


Figura 7.1. Estados por los que pasa una HU dentro de Azure DevOps.

- **New:** Este es el estado inicial de la HU al ser creada por el cliente.
- **Approved:** La HU pasa al estado approved cuando el cliente y el equipo de trabajo estén alineados con lo que se debe hacer dentro de la HU y que se defina su viabilidad. Si el desarrollo es viable, se cambia de estado y es asignada al líder del equipo.
- **Committed:** Este estado es asignado a la HU cuando se define el encargado del desarrollo dentro del equipo. El líder le asigna la HU al recurso y este se encarga de estimar el tiempo que se va a tardar en realizarla y las tareas que serán realizadas para completar la HU.

La estimación de tiempos de ejecución o “efforts” (puntos de esfuerzo en español), se definen dependiendo del tiempo que requiera el desarrollador para completar todas las tareas que contenga la HU. Estos efforts deben cumplir con la serie de Fibonacci (1, 2, 3, 5, 8) y cada uno representa un día de trabajo, es decir, 8 horas de desarrollo.

El recurso define también las tareas que deberá realizar para completar a cabalidad lo que se solicita dentro de la HU, estas tareas se deben añadir al momento en el que el requerimiento pasa al estado Committed. Las

tareas poseen la estructura que se muestra en la figura 7.2 y que es explicada a continuación:

1. **Nombre de la tarea:** Nombre con el que será identificada la tarea.
2. **Estado:** Estado en la que se encuentra actualmente la tarea. Existen 3 posibles estados: To Do, In Progress o Done (ver figura 7.3).
3. **Actividad:** El tipo de actividad que se va a realizar en la tarea. Para el caso del equipo de trabajo siempre era referido como “Development”.
4. **Trabajo estimado:** Cantidad de horas que estimaba el desarrollador en realizar la tarea.
5. **Trabajo verdadero:** Cantidad de horas reales que tardó el desarrollador en terminar la tarea.
6. **Trabajo restante:** Cantidad de horas restantes para terminar la conclusión de la tarea.

Figura 7.2. Estructura de tarea creada por el desarrollador dentro de la HU.

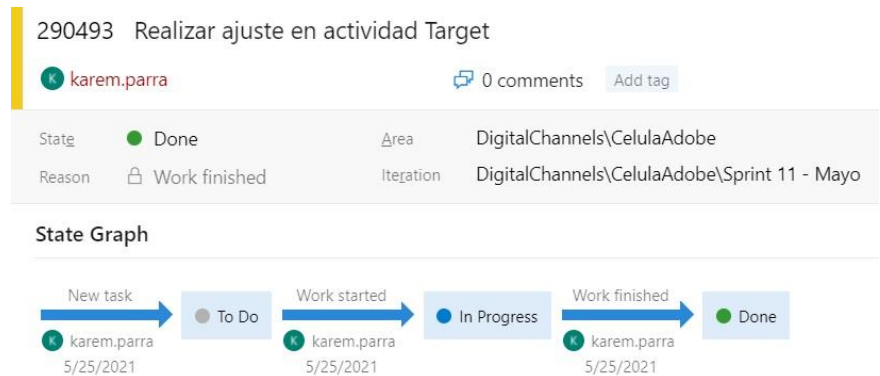


Figura 7.3. Flujo de estados de una tarea.

- **CodeReview:** Una vez el recurso haya finalizado el desarrollo, completado las tareas, se haya subido el código fuente al repositorio del proyecto y se haya lanzado el desarrollo al ambiente pre-productivo, se pasa la HU a este estado para que la persona encargada del repositorio de su revisión y aprobación para continuar con el proceso.
- **QA:** Este estado no se utiliza en el flujo de trabajo de la célula Adobe, así que la HU pasa por él pero no tiene relevancia dentro del desarrollo de ella.
- **Build:** Como la herramienta que se utiliza para el despliegue de los desarrollos inserta el código por medio de un script en la página, este estado no se requiere dentro del flujo de la HU, se pasa por él pero no se realiza ninguna validación.
- **UAT:** En este estado se realizan las pruebas de aceptación de usuario (User Acceptance Testing por sus siglas en inglés). Se le asigna la HU a un tester del lado del cliente, quien prueba el desarrollo en ambiente pre-productivo y si cumple con todos los criterios de aceptación se continúa con el siguiente estado.



- **Ready4Deployment:** Cuando las pruebas por parte del cliente son aceptadas, se pasa la HU a este estado y se reasigna al desarrollador inicial para que despliegue la actividad o el desarrollo en ambiente productivo. En este punto se informa y se reasigna al mismo tester para que pruebe el correcto funcionamiento y visualización del desarrollo en producción.
- **Done:** Una vez el desarrollo esté probado y aprobado en ambiente productivo por parte del tester del cliente, este cambia el estado de la HU a Done, donde finaliza el flujo de trabajo.
- **Removed:** Este estado es usado únicamente cuando se decide no continuar con la HU, puede ser utilizado en cualquier momento del flujo.
- **Revisión y localización de componente a ser modificado.**

Cuando se termina de estimar el effort requerido para la HU y se crean las tareas necesarias, se procede a revisar el componente que necesita ser modificado.

Debido a que los desarrollos consisten en la inserción de un script para modificar el contenido y los estilos de los elementos, es necesario buscarlo dentro del DOM, para tomar de él el identificador y conseguir el cambio necesario.

- **Desarrollo del código.**

Una vez se haya identificado el elemento a modificar, se procede a realizar el cambio solicitado. Para estos desarrollos se utiliza CSS, JavaScript y JQuery ya que son cambios simples de sus atributos. En la figura 7.4 se muestra un código corto creado con JavaScript y CSS, en el cual se inserta un nuevo texto a una de las páginas dadas unas condiciones específicas para que se muestre o no.

```

(function(){
  var tarifaIndia = digitalData.product[0].productInfo.productDetails.flightTiket.bound;
  let texts = {
    es: "Si recibiste un ascenso de cabina, tu franquicia de equipaje es de 2 maletas facturadas de 32 kgs cada una en aeropuerto y una maleta d
    en: "If you recieved an upgrade to business, you are allowed two pieces of checked baggage weighing up to 32kg each to check at the airport,
    pt: "Se você recebeu um upgrade de cabine, sua franquia de bagagem é de 2 malas despachadas de 32 kg cada no aeroporto e uma de mão."
  }
  let _convertLang = [
    es: 'es',
    en: 'en',
    us: 'en',
    pt: 'pt',
    po: 'pt'
  ];

  var _lang = _convertLang[eBaDataLayer.language.toLowerCase()];
  let container = document.querySelectorAll('.baggage-policy-for-bound');
  let elem = document.querySelectorAll('span.cabin-bag-title-label');
  for(var i = 0; i < tarifaIndia.length; i++){
    if(tarifaIndia[i].segment.find(e => e.bookingClass == "I")){
      let clone = elem[i].cloneNode();
      let newE = document.createElement('div');
      newE.classList.add("cabin-bag-title-label");
      newE.classList.add("baggage-line");
      newE.classList.add("disc-ascen");
      clone.innerHTML = texts[_lang];
      newE.appendChild(clone);
      container[i].append(newE);
    }
  }
})();

.cabin-bag-title-label.baggage-line.disc-ascen > span {
  font-weight: normal !important;
  margin-left: 0px !important;
}

body[data-page-device-type="DESKTOP"] .cabin-bag-title-label.baggage-line.disc-ascen{
  margin-left: 18px !important;
  margin-right: 16px;
}

body[data-page-device-type="MOBILE"] .cabin-bag-title-label.baggage-line.disc-ascen
{
  margin-left: 9px !important;
  padding-right: 12px !important;
}

```

Figura 7.4. Código JS y CSS que da cumplimiento a un requerimiento del cliente.

- **Creación de actividad en Adobe Target.**

Adobe Target es la herramienta que se usa para insertar el script que contiene el código dentro de la página requerida. Estos desarrollos se despliegan en ambiente pre-productivo y productivo por medio de las “actividades” que se crean dentro de la herramienta. Para cada una de esas actividades se define o configura la página o páginas donde debe insertarse el script y los usuarios que podrán visualizarlos. Para realizar la acotación de los usuarios, se utilizan códigos que filtran diferentes escenarios presentes en la página, como por ejemplo: nombre de la página, POS o país, idioma o ambiente de despliegue.

Adobe Target presenta estos códigos en forma de bloques que pueden ser insertados y combinados dentro de la actividad de forma que sea más sencillo para

el desarrollador limitar la experiencia del usuario dentro de la página (ver figura 7.5).

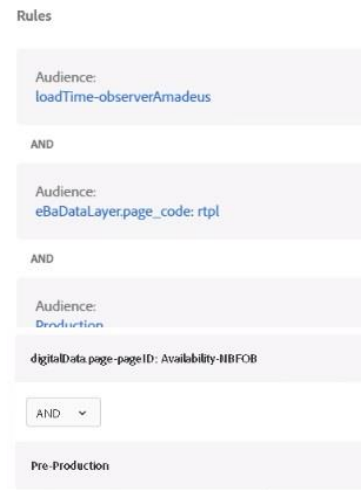


Figura 7.5. Ejemplo de algunas audiencias que se pueden encontrar en Adobe Target, las listadas se relacionan con el nombre de la página (PageID o page\_code) y el ambiente donde se despliegan las actividades (Production o Pre-Production).

En primer instancia, la actividad es configurada con la audiencia de pre-production para que se visualice dentro de la página donde se realizarán las UAT's. Por último, se deja el comentario dentro de la HU que el desarrollo fue realizado y que está listo para ser probado en QA.

- **Pruebas de funcionamiento en ambiente pre-productivo por parte del cliente.**

Cuando el desarrollo es realizado y confirmado, pasa por todos los estados hasta llegar al de UAT. En este punto se asigna la HU al tester designado por el cliente, quien verifica que todos los criterios de aceptación se estén cumpliendo. Si el desarrollo pasa todas las pruebas, se reasigna al desarrollador y se cambia el estado de la HU para continuar con el siguiente paso, si no, de igual manera se le reasigna al desarrollador pero no cambia de estado hasta que no cumpla todos los requisitos.

Dependiendo del fallo que se presente en el desarrollo y el tiempo que el recurso asignado considere que se demora en solucionarlo, se pueden reportar los errores

directamente dentro de la HU o se pueden crear y vincular a ella otro tipo de tareas que se definen como “bug”, las cuales son utilizadas para todas las pruebas y soluciones presentadas para ese error en específico.

- **Paso de implementación del desarrollo al ambiente productivo.**

Una vez la HU haya pasado al estado Ready4Deployment, el desarrollador cambia la audiencia del ambiente en la actividad en Target (de Pre-Production a Production), realiza pruebas de funcionamiento, informa dentro de la HU el paso a producción de la actividad y asigna la HU al tester del cliente, sin cambiar el estado de esta.

- **Pruebas de funcionamiento en ambiente productivo.**

El último paso requerido para que la HU sea finalizada y aprobada es la revisión por parte del cliente del funcionamiento de la actividad en el ambiente productivo. Si el desarrollo cumple con todos los criterios de aceptación, el tester cambia el estado de la HU a Done e informa dentro de la herramienta que el desarrollo fue finalizado con éxito. Si el tester logra identificar algún error de la actividad o si está generando problemas con los demás componentes de la página, este le debe informar al desarrollador para que se baje el desarrollo de producción lo más pronto posible y así, crear una actividad tipo bug (como la que se explicó en el paso de “Pruebas de funcionamiento en ambiente pre-productivo por parte del cliente”) para que el recurso pueda solucionar los errores sin que se vea afectada la interacción de los usuarios con la página.

Otro tipo de requerimientos que se atendían dentro del equipo para el proyecto A eran los que estaban relacionado con la herramienta Adobe Campaign Classic. Con esta, el cliente administra todos los temas referentes al marketing digital de su marca por medio de canales email, web y móvil. Estas campañas de marketing se dividen en dos grupos básicos:

- **Envío de e-mails:** Por medio de este canal se envían a los usuarios (con previo consentimiento) correos con promociones, beneficios, url's para la continuación de alguna compra que se estuviera haciendo por medio de la página pero que no se finalizó y otros tipos de información de importancia para el cliente.
- **Notificaciones push:** El cliente también brinda el servicio de notificaciones push para app y para el sitio web (con previo consentimiento del usuario). De estas también se derivan dos tipos:
  - Push Promocionales: Estas se envían tanto en app como en web. Notifican acerca de promociones del servicio.
  - Push Operativas: Este tipo de push son enviadas únicamente por medio de la app. Notifican al usuario acerca de cambios en el itinerario del servicio, de rutas, cancelación del servicio, recordatorios de horarios, etc.

Debido a que el tema de las notificaciones push es un tema crítico para el cliente ya que con estas, en específico con las de tipo operacional, se le informa a los usuarios sobre cambios en el servicio brindado en tiempo real, nunca se tuvo acceso por parte del practicante al ambiente de producción en Adobe Campaign. Sin embargo, se apoyó al equipo de trabajo en varias HU's de este tipo y en especial en una donde la primera instancia del desarrollo era en ambiente pre-productivo.

Este requerimiento fue creado debido a que el espacio en el servidor donde se alojan todos los datos de este servicio se estaba agotando, por ello se solicitó crear un flujo de trabajo (conocido dentro de la herramienta como “workflow”) que permitiera el borrado de manera periódica a unas tablas que se llenaban debido a la creación de las notificaciones push. Como la información contenida en estas tablas es sensible y de importancia para el negocio por temas de analítica de datos, se decidió que se borrarán x cantidad de registros una vez al día y que estos tuvieran una fecha de creación superior a un año a partir de la fecha en la que se iniciaba la ejecución del workflow. En la figura 7.6 se muestra el flujo

de trabajo creado para dar cumplimiento a este requerimiento en ambiente pre-productivo. Es importante aclarar que la primera actividad del flujo, encargada de programar el momento en el que se va a ejecutar, no se ajustó de acuerdo a los requerimientos del cliente debido a que era solo un flujo de prueba, dentro del ambiente productivo se ajustó a los tiempos que se acordaron con negocio.



Figura 7.6. Workflow para limpieza de tablas en ambiente pre-productivo.

Una vez probado el workflow y realizada la verificación de las funcionalidades acordadas, se implementó de igual manera en ambiente productivo, donde fue probado en sus primeras 3 ejecuciones con una cantidad de registros menor a la acordada para reducir el impacto de un posible error a causa de su desarrollo.

Ya que el contrato con el proyecto A terminó el 31 de Mayo del 2021, el practicante fue reasignado dentro de un nuevo proyecto, en el cual ejecutaría labores hasta el día final de su practica. Para este nuevo cliente, el proyecto B que se está desarrollando es la migración de datos de un sistema de información a uno nuevo, en este caso SAP. Este proyecto tiene una estimación de 1 año y 6 meses de duración y su objetivo es migrar datos de hasta dos años de antigüedad.

El equipo al que fue asignado el practicante, equipo Data Cleansing, se encarga del procesamiento de los datos a ser migrados, donde se valida su consistencia, correctitud y se busca evitar la duplicidad de los registros, para conseguir una correcta inserción en el sistema destino. Cada 2 a 3 días, llega uno o dos sets de datos para procesar. A continuación se presenta el paso a paso que se lleva a cabo con cada uno:

- **Revisión de disponibilización de set de datos e información del mismo en servidor.**

En esta actividad del proyecto de migración de datos interfieren 3 actores:

- **Empresa A:** Contratistas que se encargan de las labores de extracción de los datos desde el sistema origen.
- **Negocio:** Empresa a la cual se le está brindando el servicio de migración. Dentro de ellos se destacan dos actores: Los que se encargan de el monitoreo del avance del proyecto (administrativos) y los encargados de revisar el procesamiento de todos los datos y velar por que se pasen la mayor cantidad de datos posibles en cada ciclo de migración (operativos).
- **Accenture:** Encargados de realizar la limpieza de los datos y de la extracción e ingreso de ellos al sistema destino SAP.

En primer lugar, la empresa A genera la extracción de los datos del sistema origen. Estos son subidos a una carpeta de un servidor disponibilizado para que el equipo de Data Cleansing pueda tomarlos y procesarlos. Para cada uno de los objetos (tablas en PostgreSQL) se entrega el set de datos en un archivo y la cantidad de datos extraídos y el tiempo que se tardaron en otro archivo conocido como .log, ambos con extensión .txt. En la figura 7.7 se muestra la disponibilización de los dos archivos dentro de una de las carpetas del servidor para un objeto en específico.

Filename	Filesize	Filetype	Last modified	Permi...	Owr
..					
FileLoad_CONJC_CO_G.load	1.619	LOAD ...	1/06/2021 12:02:5...	-rwxr...	S00
Z_CONJ_CONTRATO_20210528100009.txt	35.268	Text D...	1/06/2021 11:27:2...	-rw-r-...	S00
Z_CONJ_CONTRATO_20210528_100009_9334.log	232	Text D...	1/06/2021 11:27:2...	-rw-r-...	S00

Figura 7.7. Muestra de los archivos extraídos por parte de la empresa A.

Cuando estos archivos son entregados se envía un correo por parte de la empresa A al equipo de Data Cleansing, donde se envía una evidencia de los archivos subidos y el path correspondiente a la carpeta donde estos se encuentran.

Los correos para informar el estado de cada objeto es un tema de gran importancia para el desarrollo del trabajo, ya que con estos negocio mantiene al tanto del avance del proyecto y para ambos contratistas es una evidencia del avance de sus propias tareas.

- **Ejecución y procesamiento del set de datos.**

Para el desarrollo del procesamiento de cada objeto se tienen carpetas dentro del mismo servidor compartido, divididas por el ciclo de migración, las cuales contienen a su vez una carpeta separada para cada uno de los objetos. Estas carpetas contienen la misma estructura (ver figura 7.8):

- **Carpeta Load:** A esta carpeta se transfieren los archivos disponibilizados por la empresa A.
- **Carpeta Out:** Dentro de esta carpeta se agregan los archivos que dan como resultado de cada uno de los procesos que se realizan a los objetos.
- **Shells y sql's de cada proceso:** existe un archivo .sh y .sql que se ejecutan para cada uno de los procesos que se le realizan a los datos.

Filename	Filesize	Filetype	Last modified	Permi...	Owner/Gr...
..					
load		File fol...	1/06/2021 12:02:5...	drwxr...	ifrdcco DC...
out		File fol...	8/07/2021 2:44:52...	drwxr...	ifrdcco DC...
01loader_enel_portion.load	1.203	LOAD ...	18/05/2021 7:20:0...	-rwxr...	ifrdcco DC...
00transfer_enel_portion.sh	810	Shell S...	18/05/2021 7:20:0...	-rwxr...	ifrdcco DC...
01loader_enel_portion.sh	2.378	Shell S...	18/05/2021 7:20:0...	-rwxr...	ifrdcco DC...
02validate_enel_portion.sh	1.926	Shell S...	18/05/2021 7:20:0...	-rwxr...	ifrdcco DC...
03transform_enel_portion.sh	1.568	Shell S...	18/05/2021 7:19:5...	-rwxr...	ifrdcco DC...
04extract_enel_portion.sh	2.346	Shell S...	28/05/2021 3:45:3...	-rwxr...	ifrdcco DC...
02validate_enel_portion.sql	363	SQL File	18/05/2021 7:19:5...	-rwxr...	ifrdcco DC...
03transform_enel_portion.sql	308	SQL File	18/05/2021 7:19:5...	-rwxr...	ifrdcco DC...
04extract_enel_portion.sql	364	SQL File	18/05/2021 7:19:5...	-rwxr...	ifrdcco DC...



Figura 7.8. Ejemplo de la estructura de la carpeta donde se realiza el procesamiento de un objeto.

Para el procesamiento de los objetos se utilizan cinco procesos:

- **Transferencia de archivos.**

En este proceso se ejecuta el shell con identificador inicial 00transfer. La acción que se realiza con este .sh es la de buscar todos los archivos que se alojan en la carpeta señalada por la empresa A y transferirlos a la carpeta correspondiente del objeto donde se va a realizar el procesamiento.

- **Carga de registros.**

En este proceso se ejecuta el shell con identificador inicial 01loader. Este .sh se encarga de cargar todos los registros que vienen dentro del archivo .txt en la tabla de nombre “input” dentro de la base de datos del equipo Data Cleansing para poder realizar los demás procesos. Esta tabla posee una estructura concertada y definida entre negocio y la empresa A antes de empezar cualquier ciclo de migración. Si alguno de los campos del registro que se va a cargar no cumple con las condiciones establecidas dentro de la tabla, este se descarta, no se carga en la tabla y se escribe dentro de un archivo de salida, con nombre loader.log, la primary key del registro que no cargó y el motivo del rechazo. Este archivo de salida se utiliza para informar a negocio y a la empresa A sobre el error generado o la novedad de cargue como se conoce dentro del proyecto.

En una primera ejecución del cargue, si se presentan novedades se informan por medio de un correo a los operativos del negocio y a la empresa A, tanto el porcentaje de registros que presentan las novedades, como la razón del por qué se presentaron. En conjunto se define donde se presentó el problema, ya sea en la extracción por parte de la empresa A o en la configuración de la tabla input por parte de Data Cleansing y también se decide si se continúa con el procesamiento, que cambios se deben

realizar para solucionar el error y si se debe enviar nuevamente la extracción del objeto.

Para realizar el informe de estas novedades, es utilizada una aplicación adicional desarrollada por uno de los integrantes del equipo. Dentro de está se sube el archivo loader.log y la aplicación lo procesa para crear y entregar un archivo excel que cuenta con dos hojas: una donde se observa la primary key del registro con novedad, la razón por la que se presentó la novedad y el .txt al cual pertenece el registro y otra que contiene el resumen de cuantos registros no se cargaron en total y también la discriminación de ellos por tipo de novedad y set de datos al cual pertenecen. En la figura 7.9 se presentan ejemplos del archivo de reporte de novedades de cargue.

Archivo Origen	Detalle log error	Tipo
DOCUMENTS_CASTIGADA_20210618123142.txt	DETAIL: Key (zopbel_fkko, opupk_fkko, tipo_record, id_cargamento, id_job)=(C100468125, 1, G, 5, 7) already exists.	Llave duplicada
DOCUMENTS_CASTIGADA_20210618123142.txt	DETAIL: Key (zopbel_fkko, opupk_fkko, tipo_record, id_cargamento, id_job)=(C111469555, 1, G, 5, 65) already exists.	Llave duplicada
DOCUMENTS_CASTIGADA_20210618123142.txt	DETAIL: Key (zopbel_fkko, opupk_fkko, tipo_record, id_cargamento, id_job)=(C112456545, 2, G, 5, 33) already exists.	Llave duplicada
DOCUMENTS_CASTIGADA_20210618123142.txt	DETAIL: Key (zopbel_fkko, opupk_fkko, tipo_record, id_cargamento, id_job)=(C112815641, 2, G, 5, 44) already exists.	Llave duplicada
DOCUMENTS_CASTIGADA_20210618123142.txt	DETAIL: Key (zopbel_fkko, opupk_fkko, tipo_record, id_cargamento, id_job)=(C113894859, 1, G, 5, 89) already exists.	Llave duplicada
DOCUMENTS_CASTIGADA_20210618123142.txt	DETAIL: Key (zopbel_fkko, opupk_fkko, tipo_record, id_cargamento, id_job)=(C114008244, 3, G, 5, 35) already exists.	Llave duplicada

Archivo Origen	Tipo	Count of Tipo
DOCUMENTS_CASTIGADA_20210618123142.txt	Llave duplicada	21
<b>Suma total</b>		<b>21</b>

Figura 7.9. Ejemplo de archivo generado por novedades de validación.

- **Validación de los registros.**

En este proceso se ejecuta el shell con identificador inicial 02validate y se usa el .sql con el mismo nombre. Con este proceso se realizan una serie de validaciones de los registros cargados contra los cargados de otros objetos, todo esto para verificar la consistencia y las relaciones que existen entre los objetos y que se deberían mantener entre ellos.

Al igual que en el proceso de cargue, dentro del proceso de validación pueden salir novedades que deben ser informadas a la parte operativa del negocio y a la empresa A, para que se definan las soluciones que se deben realizar y cómo debe continuar el procesamiento del objeto. Los registros y las diferentes novedades que se presenten son insertadas dentro de la tabla “log”. Un solo registro puede tener varias novedades asociadas, así que la estructura del registro que será insertado dentro de la tabla log contendrá la primary key del registro que presenta la novedad más una serie de campos asociados al tipo de novedad y su descripción. Debido a que estas novedades y su descripción están presentes en la tabla log, desde el aplicativo se puede obtener el reporte sin necesidad de utilizar un archivo adicional que contenga la información de estas novedades.

El reporte generado contiene dos archivos importantes: uno que contiene un resumen del id de la novedad, su descripción, cuantos registros cuentan con esa novedad, el porcentaje de registros que presentan esta novedad y la cantidad de registros totales que presentan al menos una novedad y, el otro archivo (que pueden ser varios en función de la cantidad de tipos de novedades que se generen) contiene todos los registros que incurrieron en esa misma novedad. En la figura 7.10 se muestra un ejemplo del reporte y los archivos generados.

COD_ERROR	DESCRIPCIÓN ERROR	ZOPBEL_FKKKO	OPUPK_FKKOP	TIPO_RECORD	ID_CARICAMENTO	ID_JOB	XBLNR_FKKKO
C117	Numero Documento Referencia de FKKKO (XBLNR) existe en DOCUMENTS	C632529748	3	G	5	31	262097351
C117	Numero Documento Referencia de FKKKO (XBLNR) existe en DOCUMENTS	C229947347	3	G	5	41	221861724

COD_ERROR	DESCRIPCIÓN ERROR	CANTIDAD	% ERROR
D208	Valor Invalido Dominio de Control - Campo: Operacion Parcial Doc (TVORG)	6.165	0,696%
V166	Importe Moneda Transacción Signo +/- de FKKOP (BETRW) es igual a cero	9	0,001%
C117	Numero Documento Referencia de FKKKO (XBLNR) existe en DOCUMENTS	2	0,0%
	TOTAL NOVEDADES	6.176	
	Registros con novedades	6176	

Figura 7.10. Ejemplo de archivo generado por novedades en validación.

- **Transformación de los registros.**

En este proceso se ejecuta el shell con identificador inicial 03transform y se usa el .sql con el mismo nombre. Con la ejecución de este shell se realiza la inserción de los registros dentro de la tabla “output” que lograron ser cargados en la tabla input y que no presentaron novedades de validación.

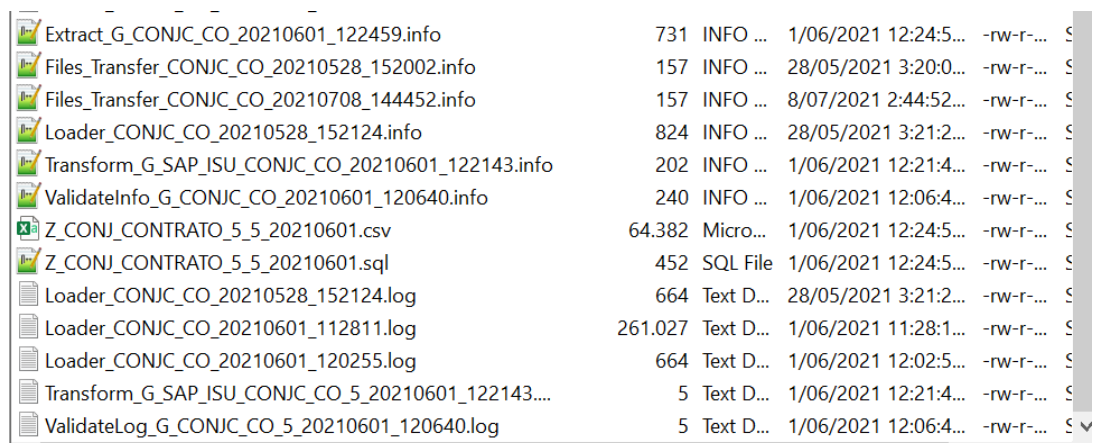
- **Extracción de los registros.**

En este proceso se ejecuta el shell con identificador inicial 04extract y se usa el .sql con el mismo nombre. En este último paso, se toman los registros de la tabla output y se genera un archivo con extensión .csv que los contiene. Este archivo es escrito dentro de la carpeta output desde la cuál puede ser extraída por parte de los encargados de la carga de los registros en SAP.

Como se mencionó anteriormente, cada uno de los procesos genera unos archivos de salida que son escritos dentro de la carpeta out de cada objeto (ver figura 7.11). Estos archivos tienen una funcionalidad específica para los diferentes procesos que se llevan a cabo dentro del equipo:

- **Archivos .info:** Estos archivos son generados por los 5 procesos. Contienen datos relacionados al ciclo de migración, identificador del objeto que fue procesado y el tiempo de inicio y fin del proceso. Estos archivos son usados en el aplicativo de reporting para generar un reporte de tiempos presentado en el informe final de resultados a negocio.
- **Archivos .log:** Estos archivos son generados por todos los procesos, excepto por el proceso de transferencia. Contienen el estado del proceso y los errores que se generan dentro de él.

- **Archivo .csv:** Este archivo se genera en el proceso de extracción y es el que contiene los registros que pueden ser migrados hacia el sistema destino.
- **Archivo .sql:** Este archivo también se genera en el proceso de extracción y contiene la consulta que se realizó para obtener los registros que pueden ser migrados al sistema destino.



Extract_G_CONJC_CO_20210601_122459.info	731	INFO ...	1/06/2021 12:24:5...	-rw-r-...	S
Files_Transfer_CONJC_CO_20210528_152002.info	157	INFO ...	28/05/2021 3:20:0...	-rw-r-...	S
Files_Transfer_CONJC_CO_20210708_144452.info	157	INFO ...	8/07/2021 2:44:52...	-rw-r-...	S
Loader_CONJC_CO_20210528_152124.info	824	INFO ...	28/05/2021 3:21:2...	-rw-r-...	S
Transform_G_SAP_ISU_CONJC_CO_20210601_122143.info	202	INFO ...	1/06/2021 12:21:4...	-rw-r-...	S
ValidateInfo_G_CONJC_CO_20210601_120640.info	240	INFO ...	1/06/2021 12:06:4...	-rw-r-...	S
Z_CONJ_CONTRATO_5_5_20210601.csv	64.382	Micro...	1/06/2021 12:24:5...	-rw-r-...	S
Z_CONJ_CONTRATO_5_5_20210601.sql	452	SQL File	1/06/2021 12:24:5...	-rw-r-...	S
Loader_CONJC_CO_20210528_152124.log	664	Text D...	28/05/2021 3:21:2...	-rw-r-...	S
Loader_CONJC_CO_20210601_112811.log	261.027	Text D...	1/06/2021 11:28:1...	-rw-r-...	S
Loader_CONJC_CO_20210601_120255.log	664	Text D...	1/06/2021 12:02:5...	-rw-r-...	S
Transform_G_SAP_ISU_CONJC_CO_5_5_20210601_122143....	5	Text D...	1/06/2021 12:21:4...	-rw-r-...	S
ValidateLog_G_CONJC_CO_5_5_20210601_120640.log	5	Text D...	1/06/2021 12:06:4...	-rw-r-...	S

Figura 7.11. Archivos generados durante el procesamiento de un objeto.

- **Entrega de informe sobre el procesamiento de los datos.**

Como paso final, se realiza la entrega del informe del proceso realizado a todos los actores que intervienen en esta parte del proyecto. Este debe ser enviado por cada objeto que se entregue para ser procesado y por cada entrega o iteración del objeto que se envíe o se suba al servidor. Dentro de su estructura, el informe debe contener:

- La cantidad de registros recibidos.
- La cantidad y una evidencia de los registros cargados en la base de datos del equipo Data Cleansing.
- El porcentaje de registros cargados en comparación a los entregados.

- El resumen de los tiempos de ejecución de los 5 procesos más el tiempo que se tardó la extracción del objeto del sistema origen por parte de la empresa A.
- El tiempo total que se llevó al realizar todos estos procesos.
- Un reporte y cantidad de las novedades de carga y validación, si se presentaron.
- Porcentaje de aceptación de la migración del objeto.

Para cada ciclo de migración, se define un indicador de aceptación del procesamiento de los objetos. Este indicador se refiere al porcentaje de registros que logran ser migrados al sistema destino sobre los que fueron extraídos del sistema origen . Para el ciclo actual el porcentaje de aceptación para los registros migrados era del 90%, si para un objeto se conseguía un porcentaje de migración menor, se dice que el proceso termina en estado KO y no se realiza la extracción de este dentro de Data Cleansing para que se pueda subir a SAP.

## 8. CONCLUSIONES

Al terminar el proceso de prácticas dentro de la empresa Accenture se obtuvieron los siguientes resultados:

- Se logró generar un valor agregado a los proyectos donde se brindó apoyo durante desarrollo de la práctica, mejorando la experiencia de usuario de los clientes y cumpliendo con las tareas requeridas para los diferentes sprints que se plantearon en el proyecto A y contribuyendo a la consecución de los objetivos de migración de datos del proyecto B.
- Dentro del proyecto A, fueron atendidas por el practicante 20 historias de usuario relacionadas con la mejora del sitio web del cliente durante los 5 meses de vinculación que se tuvo a este proyecto, además de brindar apoyo al equipo de trabajo en otras 26 HU's.
- De las HU's creadas para el proyecto A y que estaban relacionadas con la creación y actualización de workflows dentro de la herramienta Adobe Campaign Classic, se logró desarrollar 1 de ellas en ambiente pre-productivo y se apoyó al equipo de trabajo con 10 HU's de este tipo.
- Durante el mes que duró la participación del practicante en el proyecto B, se logró apoyar en todo un ciclo de migración, procesando 34 objetos y realizando varias iteraciones de procesamiento para aproximadamente 14 de ellos.
- Con el desarrollo de la práctica, se consiguió adquirir conocimientos técnicos, profesionales y de habilidades blandas debido a las actividades y tareas que fueron asignadas por el escenario de práctica Accenture.
- Se logró poner en práctica los conocimientos adquiridos a lo largo de la carrera.

## 9. BIBLIOGRAFÍA

- MDN Web Docs. (s.f.). JavaScript. Recuperado el 8 de Julio de 2021 de <https://developer.mozilla.org/es/docs/Web/JavaScript>
- MDN Web Docs. (s.f.). ¿Qué es JavaScript?. Recuperado el 8 de Julio de 2021 de [https://developer.mozilla.org/es/docs/Learn/JavaScript/First\\_steps/What\\_is\\_JavaScript](https://developer.mozilla.org/es/docs/Learn/JavaScript/First_steps/What_is_JavaScript)
- JQuery. (s.f.). ¿What is JQuery?. Recuperado el 8 de Julio de 2021 de <https://jquery.com/>
- Adobe I/O. (s.f.). Adobe Experience Manager. Recuperado el 8 de Julio de 2021 de <https://www.adobe.io/apis/experiencecloud/aem.html>
- Adobe Experience League. (s.f.). ¿Qué es Target?. Recuperado el 8 de Julio de 2021 de <https://experienceleague.adobe.com/docs/target/using/introduction/intro.html?lang=es>
- Adobe Experience League. (s.f.). Documentación de Adobe Campaign Classic v7. Recuperado el 8 de Julio de 2021 de <https://experienceleague.adobe.com/docs/campaign-classic.html?lang=es-ES>
- Microsoft Azure. (s.f.). ¿Qué es DevOps? Explicación de DevOps. Recuperado el 9 de Julio de 2021 de <https://azure.microsoft.com/es-es/overview/what-is-devops/>
- PostgreSQL. (s.f.). ¿What Is PostgreSQL?. Recuperado el 9 de Julio de 2021 de <https://www.postgresql.org/docs/14/intro-what-is.html>



- Accenture. (s.f). Sobre nuestra empresa. Recuperado el 9 de Julio de 2021 de <https://www.accenture.com/co-es/about/company-index>
- Medellín Joven, alcaldía de Medellín. (s.f). Accenture. Recuperado el 9 de Julio de 2021 de <https://www.medellinjoven.com/accenture>